Mapping data (and, more importantly, converting map data into a format that we can tidily use) requires a bit more heavy lifting behind the scenes than the core tidyverse libraries provide. We start by (if necessary) installing the sf library, which will also bring a number of dependencies with it. We’ll also install a few packages that provide some functions we might use in the conversion and mapping process.

install.packages("sf")

install.packages("rgdal")

install.packages("geojsonio")

install.packages("spdplyr")

install.packages("rmapshaper")

Then we load the packages, along with others we’ve already installed (as part of the seminar).

library(geojsonio)

library(rmapshaper)

library(rgdal)

library(tidyverse)

library(spdplyr)

library(sf)

library(socviz)

We make a function, theme\_map(), that will be our ggplot theme. It consists mostly in turning off pieces of the plot (like axis text and so on) that we don’t need.

theme\_map <- function(base\_size=9, base\_family="") {

require(grid)

theme\_bw(base\_size=base\_size, base\_family=base\_family) %+replace%

theme(axis.line=element\_blank(),

axis.text=element\_blank(),

axis.ticks=element\_blank(),

axis.title=element\_blank(),

panel.background=element\_blank(),

panel.border=element\_blank(),

panel.grid=element\_blank(),

panel.spacing=unit(0, "lines"),

plot.background=element\_blank(),

legend.justification = c(0,0),

legend.position = c(0,0)

)

}

theme\_set(theme\_map())

Next we need to get the actual map data. This is often the hardest bit. In the case of Canada, their central statistics agency provides map shape files that we can use. The Shape File format (.shp) is the most widely-used standard for maps. We’ll need to grab the files we want and then convert them to a format the tidyverse can use. You won’t be able to run the code in the next few chunks unless the files are downloaded where the code expects them to be.

So, get the data from Statistics Canada. We’re going to use this link: <http://www12.statcan.gc.ca/census-recensement/2011/geo/bound-limit/bound-limit-2011-eng.cfm>. From the linked page, choose as your options *ArcGIS .shp file*, then—for example—*Census divisions* and *cartographic boundaries*. You’ll then download a zip file. Expand this zip file into the directory in your working folder named ‘data’. Then we can import the shapefile using one of rgdal’s workhorse functions, readOGR.

canada\_raw <- readOGR(dsn = "data/gcd\_000b11a\_e", layer = "gcd\_000b11a\_e",

use\_iconv=TRUE, encoding="CP1250")

Note the options here. This information is contained in the documentation for the shape files. Now we’re going to convert this object to GeoJSON format and simplify the polygons. These steps will take a little while to run:

canada\_raw\_json <- geojson\_json(canada\_raw)

canada\_raw\_sim <- ms\_simplify(canada\_raw\_json)

Then we save the resulting GeoJSON file, which we can now work with directly from here on out:

geojson\_write(canada\_raw\_sim, file = "data/canada\_cd\_sim.geojson")

We only need to do the importing, converting, and thinning once. If you already have a GeoJSON file, you can just start here.

Read the GeoJSON file back in as an sf object. The .geojson file is included in the repository, so you can load the libraries listed above and then start from here if you want.

canada\_cd <- st\_read("data/canada\_cd\_sim.geojson", quiet = TRUE)

canada\_cd

## Simple feature collection with 293 features and 6 fields

## geometry type: MULTIPOLYGON

## dimension: XY

## bbox: xmin: -141.0181 ymin: 41.7297 xmax: -52.6194 ymax: 83.1355

## epsg (SRID): 4269

## proj4string: +proj=longlat +ellps=GRS80 +towgs84=0,0,0,0,0,0,0 +no\_defs

## First 10 features:

## CDUID CDNAME CDTYPE PRUID PRNAME rmapshaperid geometry

## 1 4609 Division No. 9 CDR 46 Manitoba 0 MULTIPOLYGON (((-97.9474 50...

## 2 5901 East Kootenay RD 59 British Columbia / Colombie-Britannique 1 MULTIPOLYGON (((-114.573 49...

## 3 5933 Thompson-Nicola RD 59 British Columbia / Colombie-Britannique 2 MULTIPOLYGON (((-120.1425 5...

## 4 4816 Division No. 16 CDR 48 Alberta 3 MULTIPOLYGON (((-110 60, -1...

## 5 5919 Cowichan Valley RD 59 British Columbia / Colombie-Britannique 4 MULTIPOLYGON (((-123.658 48...

## 6 4621 Division No. 21 CDR 46 Manitoba 5 MULTIPOLYGON (((-99.0172 55...

## 7 4608 Division No. 8 CDR 46 Manitoba 6 MULTIPOLYGON (((-98.6436 50...

## 8 4811 Division No. 11 CDR 48 Alberta 7 MULTIPOLYGON (((-112.8438 5...

## 9 4802 Division No. 2 CDR 48 Alberta 8 MULTIPOLYGON (((-111.3881 5...

## 10 5951 Bulkley-Nechako RD 59 British Columbia / Colombie-Britannique 9 MULTIPOLYGON (((-124.4407 5...

Notice the proj4string there in the metadata. We’re going to change that to Canada’s favorite projection, the Lambert Conformal Conic projection.

canada\_cd <- st\_transform(canada\_cd, crs = "+proj=lcc +lat\_1=49 +lat\_2=77 +lon\_0=-91.52 +x\_0=0 +y\_0=0 +datum=NAD83 +units=m +no\_defs")

canada\_cd

## Simple feature collection with 293 features and 6 fields

## geometry type: MULTIPOLYGON

## dimension: XY

## bbox: xmin: -2529065 ymin: 5816874 xmax: 2793734 ymax: 10396190

## epsg (SRID): NA

## proj4string: +proj=lcc +lat\_1=49 +lat\_2=77 +lat\_0=0 +lon\_0=-91.52 +x\_0=0 +y\_0=0 +ellps=GRS80 +towgs84=0,0,0,0,0,0,0 +units=m +no\_defs

## First 10 features:

## CDUID CDNAME CDTYPE PRUID PRNAME rmapshaperid geometry

## 1 4609 Division No. 9 CDR 46 Manitoba 0 MULTIPOLYGON (((-457449.9 6...

## 2 5901 East Kootenay RD 59 British Columbia / Colombie-Britannique 1 MULTIPOLYGON (((-1628202 69...

## 3 5933 Thompson-Nicola RD 59 British Columbia / Colombie-Britannique 2 MULTIPOLYGON (((-1838098 74...

## 4 4816 Division No. 16 CDR 48 Alberta 3 MULTIPOLYGON (((-988280.5 7...

## 5 5919 Cowichan Valley RD 59 British Columbia / Colombie-Britannique 4 MULTIPOLYGON (((-2253700 71...

## 6 4621 Division No. 21 CDR 46 Manitoba 5 MULTIPOLYGON (((-461578.4 7...

## 7 4608 Division No. 8 CDR 46 Manitoba 6 MULTIPOLYGON (((-500377.6 6...

## 8 4811 Division No. 11 CDR 48 Alberta 7 MULTIPOLYGON (((-1350599 73...

## 9 4802 Division No. 2 CDR 48 Alberta 8 MULTIPOLYGON (((-1372235 69...

10 5951 Bulkley-Nechako RD 59 British Columbia / Colombie-Britannique 9 ## MULTIPOLYGON (((-1921887 78...

Finally, and just because I don’t have any census-district-level data to hand at the moment, we make a vector of repeated colors to fill in the map, for decoration only, if you want to color all the census divisions.

map\_colors <- RColorBrewer::brewer.pal(8, "Pastel1")

map\_colors <- rep(map\_colors, 37)

Instead, we’ll just map the fill to PRUID, i.e. the province level. But try mapping fill to CDUID instead (the census district ID), and see what happens.

p <- ggplot(data = canada\_cd,

mapping = aes(fill = PRUID))

p\_out <- p + geom\_sf(color = "gray60",

size = 0.1) +

scale\_fill\_manual(values = map\_colors) +

guides(fill = FALSE) +

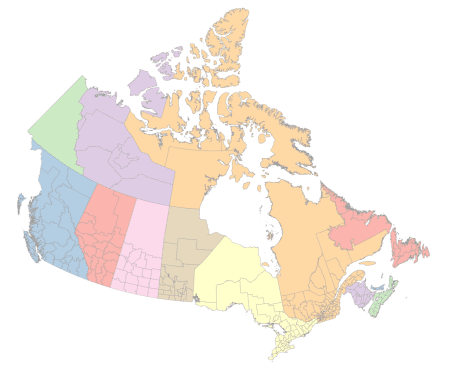
theme\_map() +

theme(panel.grid.major = element\_line(color = "white"),

legend.key = element\_rect(color = "gray40", size = 0.1))

ggsave("figures/canada.pdf", p\_out, height = 12, width = 15)

p\_out



Map of Canada, colored by Province, showing 2011 Census District boundaries.